Machine Learning
Course Details 2023

Machine Learning
mlvu.github.io
Vrije Universiteit Amsterdam

This set of slides explains the details of how the course is run, and what you need to do to pass.

|section|Course details|
|video|https://www.youtube.com/embed/XGXF_4ldzQk|

---

to pass this course

Pass **the exam**
40 points

Complete **4 quizzes**
40 points

Finish a **group project**
with a grade of 4.5 or more

Your **final grade** is the average of the project and the examination grade.
Examination = exam + quizzes.

2

Let's start with the most important information. This is what you need to do to pass.

The quizzes and the exam combine to form a single grade which is averaged with your project grade to make your final grade.

---

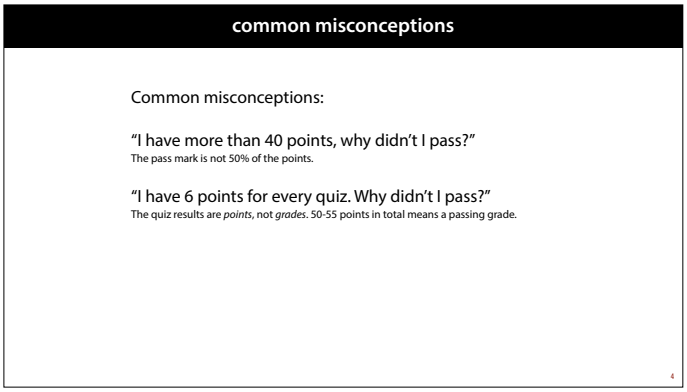
your grade

exam & quizzes	project
minimum grade 5.5	minimum grade 4.5
Exam: 40 multiple choice questions	groups of 5 no exceptions, sorry
Quizzes: open and MC questions.	subject free see Canvas for suggestions
80 pts total pass mark around 50-55 pts.	6 project sessions weekly, start second week
7 homework sessions weekly	

3

To support the exam and the quizzes, there are homework exercises. You can make these to prepare, but these are not graded. The answers are provided, so it's up to you how you use them.

You'll need to score around 50 to 55 points in total to pass the exam and quizzes. The true pass mark will be decided after the exam is completed.

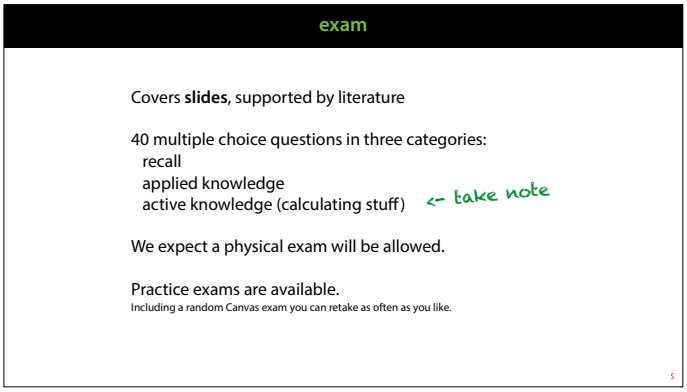


The grading system, despite our best efforts, is a little complex. This occasionally leads to some misunderstandings. These are the most common ones.

First, **scoring half the available points is not enough to pass**. There is no rule that says that half the points should correspond to a passing grade. In our case, we've found that a pass mark somewhere between 50 and 55 point for the quizzes and the exam together, means you've leanerd the minimum required subject matter. We will choose the final pass mark after the exam has been graded.

Second, the fact that you can get between 0 and 10 point for each quiz **does not mean that these represent grades**. That is, if you get 6 points for a quiz, that doesn't mean you're on your way to a *grade* of 6. The points get added to the total, and the total needs to be higher than the pass mark.

Getting 6 points for every quiz means that you'll need to score 26 points on the exam in the best case, and 31 in the worst. The latter is doable, but a relatively high score, so it's advisable to aim a little higher than this for the quizzes.



The lectures are the main focus of the exam. The literature is there as support material if the lectures aren't enough to give you a complete understanding. It's technically possible that an exam question deals with something that has only been mentioned in the literature, but it's rare.

The exam has a very predictable structure, which you can use to your advantage. See Practice Exam A on Canvas for details. The main thing to worry about are the active knowledge questions. These require you to do things like calculating something, deriving something, or following an algorithm. In short, this is the stuff you need to practice beforehand. We do so in the homework sessions.

quizzes

Weekly Canvas quiz, no time limit. Deadline on Friday.
First week is a test quiz with no grade.

Should take around an hour, if you're well prepared.

4 quizzes in total.

Includes open questions.

6

The quizzes form the other half of the examination. They are four brief assignments that you do on Canvas. They include exam-style questions, but also open questions which are manually graded by the TAs.

---

first homework this week

1 Linear Algebra

In all homework and lectures, bold lowercase letters like $\mathbf{x}$ indicate a vector, bold uppercase letters like $\mathbf{W}$ indicate a matrix and non-bold lowercase letters like x indicate a scalar (that is, a number).

Exercise 1

Explain in words what the following notations represent:

1. $f : \mathbb{R}^3 \to \mathbb{R}^2$
2. $\mathbf{y} = \mathbf{W}\mathbf{x}$
3. $\mathbf{z} = \mathbf{y}^\top \mathbf{x}$
4. $\mathbf{W} \in \mathbb{R}^{5\times 4}$

Hint: if you're not sure, see if you can find the symbols in this page: https://en.wikipedia.org/wiki/List_of_mathematical_symbols. Even

7

The homework helps you prepare for the *active knowledge* questions on the exam and in the quizzes. It's not graded, and you are entirely free to decide how much of it you do. The homework sessions, which are not obligatory, are there to help you with the homework.

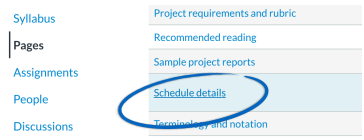
You are expected to have done the homework before showing up.

This slide shows the first homework exercise. This is **a particularly important one**, because it covers the preliminaries: the stuff we are assuming you know already. If you don't, that's fine, but it's your own responsibility to brush up. The homework has some links for where to do that.

If you find these symbols intimidating, or difficult to read, make sure to show up to the homework session. The mathematics of machine learning are relatively lightweight, but you do need to get used to the notation. If you have trouble with this sort of thing, the best thing to do is to face it early on and to ask us for help.

Homework groups:
- register on Canvas: People > Groups

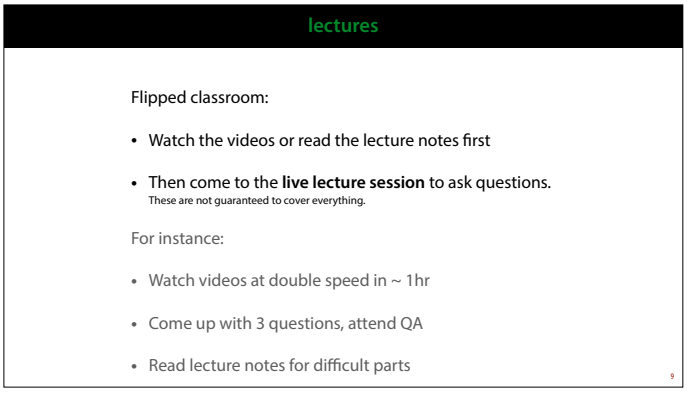
- schedule:



8

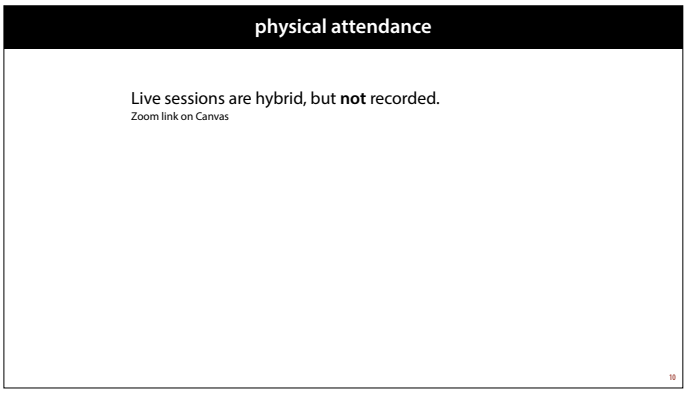
You can register for a homework group on Canvas. Go to "groups" under "People".

Go to "schedule details" under "Pages" to find out when and where every group is.



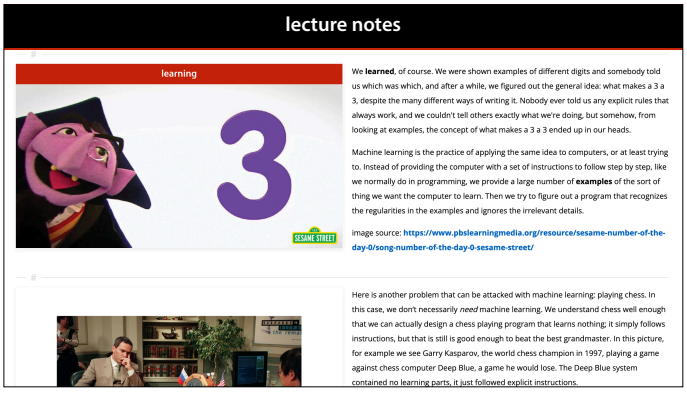
The lectures will be taught in what's know as "flipped classroom" style. That means the lectures are offered as pre-recorded videos, which you are expected to watch before the lecture. The lecture is then a question/answer session where the lecturer will go into any questions you may have, and re-explain important concepts from scratch where necessary.

You can approach this any way you like. One approach is to watch the videos at high speed, and try to understand them just well enough to formulate some questions. Then, attend the QA session, which should allow you to follow most of the discussion. After that, use the written lecture notes to work through the more complex parts slowly.



Under the current COVID regulations, we are allowed a maximum of 75 people in the lecture room.

To ensure that we don't break this limit you will have to sign up to physically attend each lecture. If we get get more than 75 signups, we will make a random selection.



The lecture notes are available in HTML as well as PDF. We also make sure that reading the notes is a proper alternative to watching the videos. You can use either medium to follow the material (and probably using both is the best option).

There will be places where the notes are a little ahead of the videos. There will be no exam/quiz questions about material that is only in the videos or only in the notes. The differences will only be in extra examples, or slightly improved explanations. **In short, you are not expected to watch all videos *and* read all lecture notes.** Either one is good enough.

We've had annotated lectures in PDF all along, but this year we're making them first-class citizens. Reading the lectures should be just as good a way to follow the course as watching the videos.

project

Until February 23

- Explore. Practice.
- Worksheets for Python stack.

On February 23: pick a topic

After February 23

- Do experiments, write report.
- NB: March 10 is the last quiz.

12

The second part of the grade is **a machine learning project**. This could be trying to analyse a particular dataset, solving a particular problem, or implementing a particular algorithm from scratch.

Our suggestion is to spend the first weeks exploring and trying out different methods. We offer 5 Jupyter notebooks (the *worksheets*) to help you get acquainted with the Python machine learning stack. Each should take 15 to 30 minutes to work through, and serve mostly to give you a working environment to play around in.

The deadline for picking a topic is 23 February. You can of course, commit to the topic earlier, so you have more time to perfect the report, but we do recommend exploring a bit first.

Note also that the last quiz is handed in on March 10, so you can expect to have a little more time to focus on the project in the last two weeks before the exam week.

---

groups: self-signup

Make your own groups.

Coordinate on the discussion board.

Please don't join empty groups.

~~1 / 5 students~~

13

You may make your own groups however you like. There is a thread on the discussion board where you can look for groups to join. It's good to try and find a group with a similar level of ambition to your own.

In principle, you are free to join any group that has space. If you don't much care, just join any group and introduce yourself.

As a courtesy to other students, please do not just join empty groups as a solo student. Join a group that is already part full. A lot of groups need a little time to coordinate and get 5 people together, before they sign up, and this becomes very difficult when all groups already have one person in them.

---

project sessions

Every week, starting next week

Informal presentations, **but** one or two slides are required.

Report on your progress, *or lack thereof.*
Describe what problems you've run into. Discuss solutions with the group. and the TA.

Discuss what subjects you're considering.

At least one group member must be present.

14

The project is supported in project sessions. There is no project session in the first week, and the project session in the last week is optional.

You'll need to give a small presentation each week. This should be very informal, but you **do need to have slides** (just one or two), to show that you've thought about what you're going to say beforehand.

Showing up without slides is counted as not showing up at all.

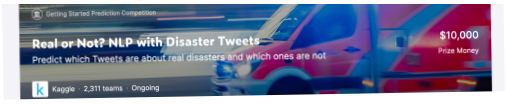
some example slides:

15

People often find it difficult to figure out what to talk about in each session, so here are some examples of the sort of slides we expect from you.

---

topics we're considering



option 1: kaggle project: Predict disaster tweets
questions:
Is this doable with basic ML? Deep learning seems too ambitious.
What if we don't get good results?

option 2: code a neural network from scratch

Too ambitious? What would our experiments consist of?

option 3: predict the results of football matches
where would we get the data?

16

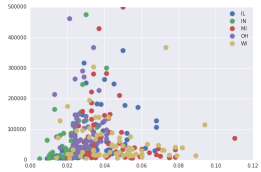
In the first week, you can introduce your group, and discuss the topics you're considering. This allows everybody to get a sense of what kind of topics different groups are thinking about, and it allows the TA to give you some feedback on what is manageable.

---

first results

success:

Managed to load the data
First plot



problems:

The data is very unbalanced.
The data seems to overlap a lot. We're not sure if the features are informative.

17

In the week after that, we have perhaps managed some early data explorations for one of your candidate topics. If so, copy paste whatever you have onto the slides, and discuss what you're stuck on and what problems you see.

Help the TA to help you do some early troubleshooting. Problems like unbalanced data, missing values or poor label quality are important to identify early.

we have a bug :(

data loading bug:

```
Traceback (most recent call last):
  File "daily.py", line 4, in <module>
    Q = pd.read_csv(StringIO("""/cygdrive/c/temp/MQM Q.csv"""), header=0, usecols=['Date-Time'
  File "/usr/lib/python2.7/site-packages/pandas-0.14.0-py2.7-cygwin-1.7.30-x86_64.egg/pandas/
    return _read(filepath_or_buffer, kwds)
  File "/usr/lib/python2.7/site-packages/pandas-0.14.0-py2.7-cygwin-1.7.30-x86_64.egg/pandas/
    parser = TextFileReader(filepath_or_buffer, **kwds)
  File "/usr/lib/python2.7/site-packages/pandas-0.14.0-py2.7-cygwin-1.7.30-x86_64.egg/pandas/
    self._make_engine(self.engine)
  File "/usr/lib/python2.7/site-packages/pandas-0.14.0-py2.7-cygwin-1.7.30-x86_64.egg/pandas/
    self._engine = CParserWrapper(self.f, **self.options)
  File "/usr/lib/python2.7/site-packages/pandas-0.14.0-py2.7-cygwin-1.7.30-x86_64.egg/pandas/
    col_indices.append(self.names.index(u))
ValueError: 'Value' is not in list
```

what we've tried:
- The error doesn't show up in google.
- CSV file does load in Excel, google sheets

18

If you get stuck somewhere, just present the problem, and what you've tried so far. Maybe some students have encountered the same problem, maybe the TA can give you some tips for how to solve the problem.

You don't always have to present progress, but you do have to do something every week, and present what you've done.

Note that it's beneficial to work in python for this sort of thing, because most other students will also be working in python, so it increases the chances that your fellow students will be able to help you.

---

our code so far

```
1   def sigmoid(Z):
2       return 1/(1+np.exp(-Z))
3
4   def relu(Z):
5       return np.maximum(0,Z)
6
7   def sigmoid_backward(dA, Z):
8       sig = sigmoid(Z)
9       return dA * sig * (1 - sig)
10
11  def relu_backward(dA, Z):
12      dZ = np.array(dA, copy = True)
13      dZ[Z <= 0] = 0;
14      return dZ;
```

19

If you're halfway through some complex coding, you can present some code that's finished, and explain what it does.

---

warning

If you don't present, **with slides**, it counts as being absent, which will hurt your grade, or cause you to fail the project (in extreme cases).

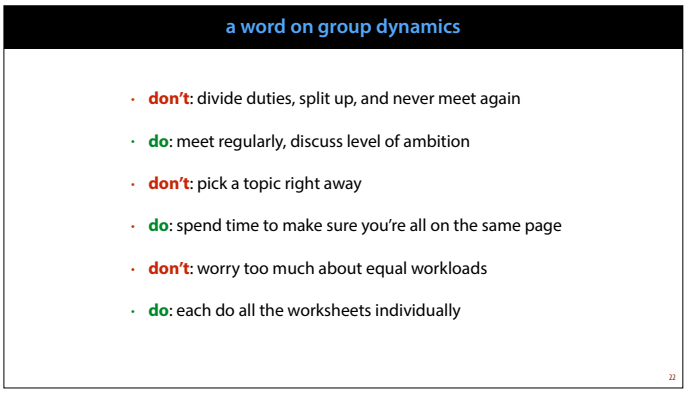
20

We won't fail you for being absent (or not presenting) just once, but we may lower your grade. If you miss *all* group meetings or never present, we may choose to fail you for the project. At the very least, we will not give you the benefit of the doubt in edge cases.

Some people don't like working in groups, so they turn a project assignment into five individual assignments by immediately breaking up the work and never meeting again. Not only is this not the point of group work, it's very likely to fail: if one group member doesn't deliver, or misunderstood the idea, the whole project goes wrong.

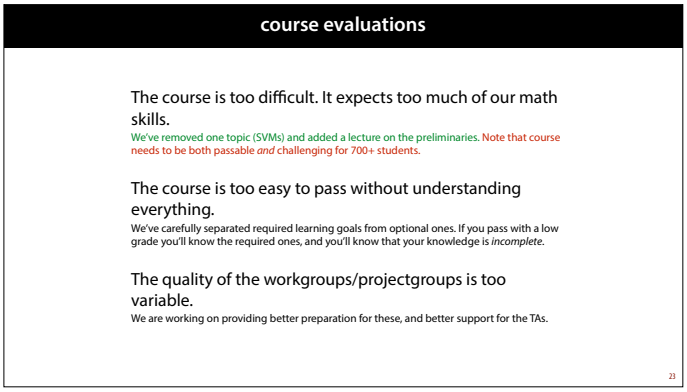
We consider efficient group work one of the skills you are practicing in this course. Just like coding, writing and mathematics. If you don't invest in setting up a healthy group dynamic, it will hurt your grade. In principle, we will not intervene if a group member underdelivers or otherwise doesn't satisfy the obligations the group has set. Unless somebody stays entirely absent, it's your own responsibility.

Please don't let that stop you, however, from letting us know about any problems. Just like the other skills, this is something we can hopefully help out with. We just want to emphasize that it's your own responsibility. We're happy to help, but not to arbitrate.



We have a very diverse group of students, and many will be grouped with people you haven't met before. Make sure that you're on the same page about the kind of project you'd like to do and how complicated you want to make stuff. Nothing kills a project faster than a single highly motivated student dragging everybody into a hugely ambitious project.

To make sure that everybody is clear about the group's goals, **take your time** before you pick the topic. Meet every week and do lots of exploratory work. give everybody time to learn what machine learning is about, and to join the discussion. It helps if everybody does the worksheets on their own, so you have a common experience to build on.



Finally, we'd like to point a few things that often show up in evaluations, and what we do about these points of criticism.

The first complaint we often hear is that the course is too difficult and requires a lot of math skills. We ask programme directors to check that students have at least *some* background in probability, calculus and linear algebra before adding this course to their program. Still for some people that means having had calculus in high school, or having to recall a linear algebra courser that wasn't very successful. We do our best to give you time to brush up on your preliminaries before we get into the hard math. This year, we've added a lecture you can check before the course, to ensure that the preliminaries are up to scratch.

The flipside of this is that we have certain goals to achieve. We don't just need to convey machine learning, many of you are going to do a master next year (or some the year after) where the mathematical level is much higher. If we hold back now, you will only suffer more next year. In short, there is a limit to how much simpler we can make the

course.

On the other side, we also hear that the course is *too simple*. More specifically, that despite all the complicated stuff we talk about, you only really need to master a small subset of the material in order to pass. It's important to realize that this is *by design*. Doing this is not cheating. We carefully choose a subset of the material as primary learning goals. Knowing these inside and out will get you a passing grade, and the better you know the rest, the closer you get to a 10. This is is how course are supposed to work. Especially a course like this, with so many students.

Finally, a recurring theme is that the workgroups and project groups don't add much, or that there is too much variation in quality between the TAs. This is a difficult problem to solve at our current scale, but we are experimenting with various techniques. These include better communication between TAs and more support for TAs in designing their sessions. We can't promise that this will improve immediately, but we are working on it.

---

course difficulty

	avg passing grade	passed
2018	7.3	78%
2019	7.6	87%
2020	7.7	95%
2021	8.3	80%
2022	7.6	78%

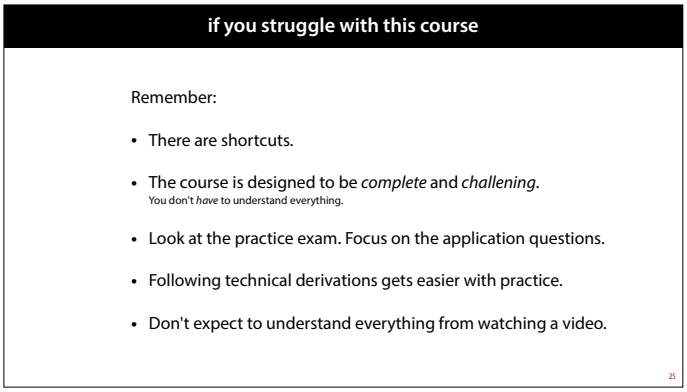
24

To illustrate the situation, here are the average passing grades and the pass percentages of previous years. Please note that these do not point to a disproportionately difficult course. In fact, some years are a little higher than what would be expected from a sufficiently challenging course.

These are the percentages of active students that passed, and the average grades among that percentage of students who passed

This doesn't mean we don't sympathise if you find the course difficult, or that we think it's your own fault. We are always happy to help, and we strongly believe everybody should be able to learn these concepts. It's just that these numbers don't indicate that there is a serious problem, or that an overhaul is called for.

Nevertheless, this year we have decided to remove some of the more technical subjects which have become less relevant in the last few years: support vector machines, and with them Lagrange multipliers. We've decided not to replace these by other subjects, but instead to give the remainder a little more room to breathe. This will hopefully lighten the load a little.



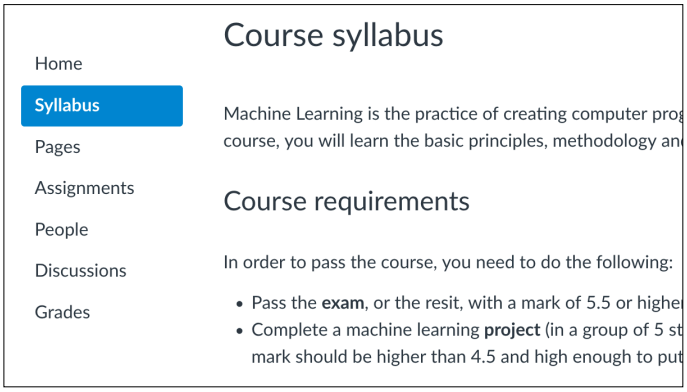
This doesn't mean you should just give up if you find yourself overwhelmed by the material. It's perfectly possible and acceptable to pass the course without understanding all the ins and outs of every topic discussed.

Quite often, the slides aim to provide a *complete* story so that if you need to know all the details about a certain topic, you have them in a self-contained package. That doesn't mean, however, that you always need to understand all details of every subject. You can often skip the technical details, so long as you understand the larger message. You don't always need to know how to get from A to B, so long as you understand what A and B are, and why it's important to get from one to the other.

To help you separate the wheat from the chaff, please look at the **practice exams** early on. They will help you to understand what it is you should focus on, if you find that the whole of a lecture is too much to digest. In general it's fine to skip the occasional technical detail, but there are a few subjects that you need to master properly for the exam.

When it comes to following complex mathematical derivations, please note that it gets easier with practice. Maybe it takes you an hour to follow all the steps in a single slide the first time around, and your heart sinks at the prospect of doing that several times for every lecture. Please note that if you persevere, it will get easier very quickly, and before long, it will be second nature.

The practice of recording videos may give the unfortunate impression that all material can be absorbed by passively watching a video. You should expect to get as much out of watching the videos passively as you get out of attending a live lecture. It should be enough to set the stage, and to give a skeletal understanding of the subject, but to get the details, an active approach is necessary: you need to watch with pen and paper ready, and you'll have to frequently pause. Hopefully, the lecture notes can help with this part of the process.



Those are the main points to understand going into the course.

If anything is still unclear, you may ask me any question you like but only after you've read this page from top to bottom. **There is an FAQ section at the bottom.**

to-do today

- Register for a workgroup
  optional: only if you plan to attend the sessions
- Get a group together, or register for a random project group.
  Or hit the **discussion board** to find a group you like
- First homework
  Check your understanding of the preliminaries
- First worksheet